ECE 509: Convex Optimization | **Rutgers: Spring 2024**

Homework #6

March 21, 2024

Name: Ravi Raghavan | *Extension: No*

Problem 1. *Pre-computation for line searches.* For each of the following functions, explain how the computational cost of a line search can be reduced by a pre-computation. Give the cost of the pre-computation, and the cost of evaluating $g(t) = f(x + t\Delta x)$ and $g'(t)$ with and without the pre-computation.

(a) $f(x) = -\sum_{i=1}^{m} \log(b_i - a_i^T x)$

Solution. In Exact Line Search, our goal is to minimize $\tilde{f}(t) = f(x + t\Delta x)$

$\tilde{f}(t) = -\sum_{i=1}^{m} \log(b_i - a_i^T(x + t\Delta x))$

$\tilde{f}(t) = -\sum_{i=1}^{m} \log(b_i - a_i^T x - a_i^T t\Delta x))$

$\tilde{f}'(t) = -\sum_{i=1}^{m} \frac{-a_i^T \Delta x}{\log(b_i - a_i^T(x + t\Delta x)))}$

$\tilde{f}'(t) = -\sum_{i=1}^{m} \frac{-a_i^T \Delta x}{\log(b_i - a_i^T x - a_i^T t\Delta x))}$

Let n be the size of each a_i and x. Let's say we are testing out k values of t.

Precomputation:

Computing $b_i - a_i^T x$ for $i \in [1, m]$ takes $2n$ operations per i.
Computing $a_i^T \Delta x$ for $i \in [1, m]$ takes $2n - 1$ operations per i

Hence, there are a total of $m(4n - 1) = O(mn)$ operations in pre-computation

Cost without Precomputation($\tilde{f}(t)$):

Computing $x + t\Delta x$ takes a total of $2n$ operations. Computing $a_i^T(x + t\Delta x)$ takes a total of $4n - 1$ operations. $b_i - (a_i^T(x + t\Delta x)))$ takes $4n$ operations. $\log(b_i - a_i^T(x + t\Delta x))$ takes $4n + 1$

$\sum_{i=1}^{m} \log(b_i - a_i^T(x + t\Delta x))$ takes $m(4n + 1)$ operations.

Taking the inverse of this sum will require $m(4n + 1) + 1$ operations

Since we are testing out k values of t, this requires $4mnk + mk + k$

Cost without Precomputation($\tilde{f}'(t)$):

Computing $x + t\Delta x$ takes a total of $2n$ operations. Computing $a_i^T(x + t\Delta x)$ takes a total of $4n - 1$ operations. $b_i - (a_i^T(x + t\Delta x)))$ takes $4n$ operations. $\log(b_i - a_i^T(x + t\Delta x))$ takes $4n + 1$. Computing $-a_i^T \Delta x$ takes $2n$ operations

Calculating $\frac{-a_i^T \Delta x}{\log(b_i - a_i^T(x + t\Delta x)))}$ takes $6n + 2$

$\sum_{i=1}^{m} \log(b_i - a_i^T(x + t\Delta x))$ takes $m(6n + 2)$ operations.

Taking the inverse of this sum will require $m(6n + 2) + 1$ operations

Since we are testing out k values of t, this requires $6mnk + 2mk + k$

Cost with Precomputation($\tilde{f}(t)$):

Computing $\log(b_i - a_i^T x - a_i^T t\Delta x))$ takes 3 operations. $\sum_{i=1}^m \log(b_i - a_i^T x - a_i^T t\Delta x))$ takes $3m$ operations. Taking the inverse of this sum will require $3m + 1$ operations

Since we are testing out k values of t, this requires $3mk + k$

Cost with Precomputation($\tilde{f}'(t)$):

Computing $\frac{-a_i^T \Delta x}{\log(b_i - a_i^T x - a_i^T t\Delta x))}$ takes 5 operations.

$\sum_{i=1}^m \frac{-a_i^T \Delta x}{\log(b_i - a_i^T x - a_i^T t\Delta x))}$ takes $5m$ operations. Taking the inverse of this sum will require $5m + 1$ operations

Since we are testing out k values of t, this requires $5mk + k$

(b) $f(x) = \log(\sum_{i=1}^m \exp(a_i^T x + b_i))$

Solution. In Exact Line Search, our goal is to minimize $\tilde{f}(t) = f(x + t\Delta x)$

$\tilde{f}(t) = \log(\sum_{i=1}^m \exp(a_i^T (x + t\Delta x) + b_i))$

$\tilde{f}(t) = \log(\sum_{i=1}^m \exp(a_i^T x + a_i^T t\Delta x + b_i))$

$\tilde{f}(t) = \log(\sum_{i=1}^m \exp(a_i^T x + b_i + a_i^T t\Delta x))$

$\tilde{f}'(t) = \log(\sum_{i=1}^m \exp(a_i^T (x + t\Delta x) + b_i)(a_i^T \Delta x))$

$\tilde{f}'(t) = \log(\sum_{i=1}^m \exp(a_i^T x + b_i + a_i^T t\Delta x)(a_i^T \Delta x))$

Let n be the size of each a_i and x. Let's say we are testing out k values of t.

Precomputation:

Computing $a_i^T x + b_i$ for $i \in [1, m]$ takes $2n$ operations per i.
Computing $a_i^T \Delta x$ for $i \in [1, m]$ takes $2n - 1$ operations per i

Hence, there are a total of $m(4n - 1) = O(mn)$ operations in pre-computation

Cost without Precomputation($\tilde{f}(t)$):

Computing $x + t\Delta x$ takes a total of $2n$ operations. Computing $a_i^T (x + t\Delta x)$ takes a total of $4n - 1$ operations. $a_i^T (x + t\Delta x) + b_i$ takes $4n$ operations. $\exp(a_i^T (x + t\Delta x) + b_i)$ takes $4n + 1$

$\sum_{i=1}^m \exp(a_i^T (x + t\Delta x) + b_i)$ takes $m(4n + 1)$ operations.

Taking the log of this sum will require $m(4n + 1) + 1$ operations

Since we are testing out k values of t, this requires $4mnk + mk + k$

Cost without Precomputation($\tilde{f}'(t)$):

Computing $x + t\Delta x$ takes a total of $2n$ operations. Computing $a_i^T (x + t\Delta x)$ takes a total of $4n - 1$ operations. $a_i^T (x + t\Delta x) + b_i$ takes $4n$ operations. $\exp(a_i^T (x + t\Delta x) + b_i)$ takes $4n + 1$. $\exp(a_i^T (x + t\Delta x) + b_i)(a_i^T \Delta x)$ takes $4n + 2$

Calculating $\sum_{i=1}^m \exp(a_i^T (x + t\Delta x) + b_i)$ takes $m(4n + 2)$ operations.

Taking the log of this sum will require $m(4n + 2) + 1$ operations

Since we are testing out k values of t, this requires $4mnk + 2mk + k$

Cost with Precomputation($\tilde{f}(t)$):

Computing $\exp(a_i^T x + b_i + a_i^T t\Delta x)$ takes 3 operations. $\sum_{i=1}^m \exp(a_i^T x + b_i + a_i^T t\Delta x)$ takes $3m$ operations. Taking the log of this sum will require $3m + 1$ operations

Since we are testing out k values of t, this requires $3mk + k$

Cost with Precomputation($\tilde{f}'(t)$):

Computing $\exp(a_i^T x + b_i + a_i^T t\Delta x)(a_i^T \Delta x)$ takes 4 operations.

$(\sum_{i=1}^m \exp(a_i^T x + b_i + a_i^T t\Delta x)(a_i^T \Delta x))$ takes $4m$ operations. Taking the log of this sum will require $4m + 1$ operations

Since we are testing out k values of t, this requires $4mk + k$

Problem 2. *True or False.*

(a) False

(b) True

(c) False

(d) True

(e) False

(f) True

(g) False